IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Applicant:	Ashley Saulsbury et al.	
App. No.:	09/992,064	Conf. No.: 4869
Filed:	November 21,2001	Art Unit: 2193
Title:	METHODS AND APPARATUS FOR PERFORMING PIXEL AVERAGE OPERATIONS	Examiner: Chat C. Do

APPEAL BRIEF

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Applicant (hereafter "Appellant") hereby submits this Appeal Brief in response to the to the final Office action mailed April 14, 2006, and further to the Notice of Appeal and the Pre-Appeal Brief Request for Review filed June 29, 2006, in the above-captioned case. This Appeal Brief is due on or before August 6, 2007, based on the Notice of Panel Decision from Pre-Appeal Brief Review mailed July 6, 2007.

Appellant respectfully requests consideration of this appeal by the Board of Patent Appeals and Interferences (hereafter the "Board").

An oral hearing is not requested at this time.

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

The invention is assigned to Sun Microsystems, Inc. of 4150 Network Circle, Santa Clara, California 95054.

II. RELATED APPEALS AND INTERFERENCES

To the best of Appellant's knowledge, there are no appeals or interferences that are related to, will directly affect, will be directly affected by, or have a bearing on the Board's decision in the present appeal.

III. STATUS OF THE CLAIMS

Claims 1-21 are currently pending in the above-referenced application. The rejection of all of the pending claims is appealed herein. No claims have been allowed. A clean copy of all claims on appeal is attached hereto as the Appendix of Claims.

IV. STATUS OF AMENDMENTS

The last Amendment submitted by Appellant was dated January 27, 2006, and was in response to the Office action dated October 28, 2005. A final Office action was entered April 14, 2006, to which no further response was submitted by Appellant. A Notice of Appeal and a Pre-Appeal Brief Request for Review were timely filed June 29, 2006.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

Independent claims 1 and 8 provide for methods for averaging two pixel values. As recited in each claim, the methods comprise an operation of decoding a single machine code instruction comprising/indicating an address for a first input register, an address for a second input register, an address for an output register, an op code indicating a function to perform, and a rounding factor. In various implementations of the claimed invention, the method may include loading a plurality of first operands from the first input register, loading a plurality of second operands from the second input register, producing, based on the op code, an average of one of the plurality of first operands and one of the plurality of second operands, and storing the average in the output register. The rounding factor, which indicates which of a plurality of rounding algorithms to use in producing the average, comprises a first rounding algorithm able to produce a change in the average and a second rounding algorithm able to produce a change in the average. The method of claim 8 recites that the first and second operands are from specific fields of the input registers and that the average is stored in a specific field of the output register.

For example, a machine code instruction (sub-instruction) 404 is illustrated in Fig. 4. The sub-instruction 404 uses the register addressing form where Rs1 is a first operand 416, Rs2 is a second operand 420, and Rd is an output operand 424. The sub-instruction 404 is divided into an address portion 408 and an op code portion 412. In this implementation, the sub-instruction 404 is thirty-two bits wide such that a four-way VLIW processor with a one hundred and twenty-eight bit wide instruction word 52 can accommodate execution of four sub-instructions 404 at a time. *See specification, page 8,lines 7-13.*

In this implementation, the sub-instruction 404 includes a first source address 416 and a second source address 420, for respective first and second registers, and a destination address 424 for a destination (output) register. *See specification, page 8,lines 17-21.* Further, the sub-instruction 404 includes op codes 412 and a rounding factor r. The rounding factor r indicates the way to round the average before being stored in the output register. The op codes 412 are used to execute a sub-instruction 54, for example, within a VLIW instruction word 52, as illustrated in Fig. 2. *See specification, page 8, lines 25-30.*

Based on the op codes 412, an instruction processor 500 may load a plurality of first operands 512-1 through 512-4 from the first input register 508-1, load a plurality of second operands 516-1 through 516-4 from the second input register 508-2, load the rounding factor (r) 512, produce an average 524-1 through 524-4 of one of the plurality of first operands and one of the plurality of second operands and store the average in the output register 504. *See Figs. 5 and 6, and specification, page 9, lines 8-33.*

Independent claim 18 provides for an apparatus for pixel averaging. As recited in the claim, the apparatus comprises a plurality of average modules respectively coupled to first and second fields of first and second input registers, wherein the modules are configured to perform an averaging function indicated by an op code in a single machine code instruction that also indicates a rounding factor. Specifically, the single machine code instruction comprises an address for the first input register, an address for the second input register, an address for an output register, the op code, and the rounding factor. In various implementations of the claimed invention, the modules are respectively coupled to the first and second fields, and to third fields of the output register. The rounding factor, which indicates which of a plurality of rounding algorithms to use in producing the average, comprises a first rounding algorithm able to produce a change in the average and a second rounding algorithm able to produce a change in the average.

For example, as discussed above, such a pixel averaging apparatus may include a first input register 508-1 comprising a plurality of fields for operands 512-1 through 512-4, a second input register 508-2 comprising a plurality of fields for operands 516-1 through 516-4, a rounding factor (r) 512, and an output register 504 comprising a plurality of fields for results 524-1 through 524-4. The apparatus may further include a plurality of average modules 600-1 through 600-4 coupled to the fields of the input registers 508-1, 508-2, and to the fields of the output register 504. *See Figs. 5 and 6, and specification, page 9, lines 8-33.*

VI. GROUNDS OF REJECTION PRESENTED FOR REVIEW

A. Whether U.S. Patent No. 6,889,242 to Sijstermans et al. (Sijstermans) discloses or teaches each and every feature of independent method claim 1 (and related dependent claims) and independent method claim 8 (and related dependent claims) so as to anticipate these claims under 35 U.S.C. § 102.

B. Whether Sijstermans discloses or teaches each and every feature of independent apparatus claim 18 (and related dependent claims) so as to anticipate this claim under 35 U.S.C. § 102.

VII. ARGUMENT

A. THE REJECTION OF INDEPENDENT CLAIMS 1 AND 8 AND THE RELATED DEPENDENT CLAIMS UNDER 35 U.S.C. § 102(e) IS IMPROPER BECAUSE THE APPLIED REFERENCE FAILS TO DISCLOSE EAH AND EVERY FEATURE RECITED IN THE CLAIMS

Method claims 1-17 are rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,889,242 to Sijstermans et al. (Sijstermans). Claims 1 and 8 are independent claims and remaining claims 2-7 and 9-17 respectively depend therefrom. The basis of rejection of the dependent claims stems from the rejection of the independent claims; accordingly, the rejection of the independent claims is addressed below.

It is respectfully submitted that the final Office action fails to set forth a proper prima facie case of anticipation of claims 1 and 8 as Sijstermans does not disclose each and every feature recited in the claims. *See MPEP § 2131*. Specifically, as argued throughout prosecution, Sijstermans fails to disclose or teach an operation of decoding a single machine code instruction comprising or indicating an address for a first input register, an address for a second input register, an address for an output register, an op code indicating a function to perform, and a rounding factor, as respectively recited in claims 1 and 8.

The basis set forth in the final Office Action for asserting that Sijstermans discloses this claimed feature is the decoder unit 18 in Fig. 1. In particular, the final Office action identifies addresses for input and output registers by citing column 3, lines 45-53, an op code by noting functional units 14 in Fig. 1 and citing column 3, lines 45-66, and a rounding factor noting 202 in Fig. 2 and citing column 4, line 55, to column 5, line 45. However, Appellant respectfully submits that this does not establish a sufficient basis for anticipation of decoding a single machine code instruction as claimed.

None of the text cited by the final Office action supports the conclusion that Sijstermans discloses a single machine code instruction including the identified registers, op code and rounding factor. While Sijstermans describes a VLIW (very long instruction word) processor able to execute various rounded averaging operations based on an instruction including register addresses and op codes, Sijstermans simply fails to disclose that the rounding factor is included in the same instruction.

The final Office Action fails to set forth any basis for asserting that the rounding factor is included in the same instruction. For example, in the Response to Arguments section, the final Office action merely asserts that "the rounding [sic] factor is included in the instruction" without citing any factual support in Sijstermans.

Appellant respectfully submits that the final Office action's reliance on Sijstermans' disclosure of a VLIW instruction is misplaced. As is well known, a VLIW packages multiple, independent instructions for multiple, independent functional units into one very long

instruction to be issued simultaneously to the functional units. This allows a processor to effectively use resources (the functional units) by performing code as scheduled by a compiler into the VLIW, to increase speed and efficiency. However, the VLIW approach does not inherently package all instructions.

While Sijstermans discloses various instructions that may be processed by a VLIW processor, such as "[if regard]avg4_bu rsrc1 rsrc2 rsrc3 rsrc4 rdest" cited by the final Office Action, none of these instructions include or identify a rounding factor. In the example cited by the final Office action, rsrc1, rsrc2, rsrc3 and rsrc4 are arguably input register addresses for operands (unsigned-byte vectors), rdest is arguably an output address for the result (averaged unsigned-byte vector), and avg4bu is arguably an op code. (Col. 5, ln. 66 – col. 6, ln. 38). Clearly, the VLIW instruction including the input and output register addresses and the op code does <u>not</u> include the rounding factor. Each of the examples disclosed by Sijstermans similarly only illustrate a VLIW instruction including input and output register addresses and an op code, <u>without any rounding factor</u>. The final Office action fails to identify any code corresponding to the rounding factor in the cited VLIW instruction, but merely asserts that the VLIW instruction includes the rounding factor.

Contrary to this unsupported assertion by the final Office action, Sijstermans discloses that the rounding factor is determined or set by executing a <u>separate</u> instruction. Each of the examples disclosed by Sijstermans, including the one cited by the final Office action, describe that the type of rounding applied depends on the value of an integer rounding mode value. (Col. 6, lns. 18-20, for example). According to Sijstermans, conventional averaging operations are supported, as described in the various examples. (Col. 4, lns. 60-61). However, a user can select from a variety of alternative rounding modes by setting the factor integer rounding mode value, i.e., setting an integer rounding mode field of a Program Control and Status Word (PCSW) register that is stored, for example, in the control register 22. (Col. 4, lns. 61-65). Thus, Sijstermans discloses that the integer rounding mode value is set by the user, causing the processor to execute a separate instruction.

Although Appellants recognize that the claims of a reference do not necessarily define the scope of the disclosure of a reference, Appellants note that the claims of Sijstermans are elucidating in this case. Each and every claim of Sijstermans specifically recites two separate instructions. For example, independent claim 1 of Sijstermans recites "executing two separate instructions . . . comprising: executing a first instruction . . . to set a rounding mode; and executing a second instruction . . . to generate an integer result . . . , wherein the second instruction does not designate the rounding mode." Independent claims 7, 11, 17, 21, 27, 31, 38, 43 and 49 recite similar limitations that clearly require a separate instruction for setting or designating a rounding mode and a separate instruction for

performing an arithmetic function or operation. Appellant respectfully submits that the final Office action improperly disregards this portion of the disclosure, which specifically teaches away from decoding a single machine code instruction as recited in Appellant's claims 1 and 8. *See MPEP § 2141.03(VI), indicating that a prior art reference must be considered in its entirety, including portions that would lead away from the claimed invention.*

When considered in light of its claims, the Detailed Description of Sijstermans becomes clearer, and should be understood as teaching two separate instructions, one for setting a rounding mode and another for carrying out the arithmetic operation(s). The Detailed Description of Sijstermans clearly supports this understanding by stating that "[o]nce a rounding mode is set, the programmable processor uses the mode when executing subsequent machine instructions that perform arithmetic operations." (Col. 2, lns. 42-45, emphasis added). Similarly, the Summary of Sijstermans supports this understanding of the Detailed Description of Sijstermans, by stating that a "machine instruction is used to set the rounding mode, which is automatically applied to subsequent arithmetic operations." (Col. 1, lns. 48-50; *see also*, col. 1, lns. 64-67).

Thus, Appellant respectfully submits that Sijstermans, when considered as a whole as required, does not disclose what is asserted by the final Office action, but actually teaches directly away from decoding a single machine code instruction as recited in Appellant's claims 1 and 8.

Accordingly for at least the reasons set forth above, it is respectfully submitted that Sijstermans does not anticipate the subject matter recited in claims 1 and 8 and the respective dependent claims. Thus, these claims are patentable over Sijstermans.

B. THE REJECTION OF INDEPENDENT CLAIM 18 AND THE RELATED DEPENDENT CLAIMS UNDER 35 U.S.C. § 102(e) IS IMPROPER BECAUSE THE APPLIED REFERENCE FAILS TO DISCLOSE EAH AND EVERY FEATURE RECITED IN THE CLAIMS

Claims 18-21 also are rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,889,242 to Sijstermans et al. (Sijstermans). Claim 18 is independent and remaining claims 19-21 depend therefrom. The basis of rejection of the dependent claims stems from the rejection of the independent claim; accordingly, the rejection of the independent claim addressed below.

It is respectfully submitted that the final Office action fails to set forth a proper prima facie case of anticipation of claim 18 as Sijstermans does not disclose each and every feature recited in this claim. *See MPEP § 2131*. Specifically, as argued throughout prosecution, Sijstermans fails to disclose or teach a plurality of average modules configured to perform an averaging function indicated by an op code in a single machine code instruction, which single machine code instruction indicates a rounding factor and comprises

an address for a first input register, an address for a second input register, an address for an output register, the op code, and the rounding factor, as recited in claim 18.

As discussed above with respect to claims 1 and 8, Sijstermans does not disclose a single machine code instruction including the identified registers, the op code and the rounding factor, and the final Office action fails to establish otherwise. It follows that Sijstermans does not disclose or teach a plurality of average modules configured as recited in claim 18.

Further, Appellant respectfully submits that the final Office action fails to establish a prima facie case of anticipation because it fails to set forth a proper basis for rejection. The final Office action rejects claim 18 by asserting that claim 18 "is an apparatus claim of claim 1." The final Office action fails to provide any basis for the alleged anticipation of the specific structures recited in claim 18, other than the rationale applied to the operations recited in claim 1. Appellant respectfully submits that this is improper and does not provide Appellant with a factual basis for the rejection, but only an implication that Sijstermans discloses such structure, in particular because claim 18 is not merely "an apparatus claim of claim 1." As the "identical invention must be shown in as complete detail as is contained in the . . . claim," Appellant respectfully submits that the stated rejection fails to establish the requisite basis for finding anticipation. *See MPEP § 2131, citing Richardson v. Suzuki Motor Co., F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).*

Thus, based at least on the arguments set forth above with respect to claims 1 and 8, Appellant respectfully submits that Sijstermans fails to disclose each and every feature of claim 18. Further, as the final Office action fails to address the specific structure recited in claim 18, Appellant respectfully submits that a proper basis for rejecting claim 18.

Accordingly for at least the reasons set forth above, it is respectfully submitted that Sijstermans does not anticipate the subject matter recited in claim 18 and the respective dependent claims. Thus, these claims are patentable over Sijstermans.

VIII. CONCLUSION

Appellant respectfully submits that all the appealed claims in this application are patentable and requests that the Board of Patent Appeals and Interferences direct allowance of the rejected claims.

Appellant believes no further fees or petitions are required. However, if any such petitions or fees are necessary, please consider this a request therefor and authorization to charge Deposit Account No. 04-1415 accordingly.

Dated: August 2, 2007

Respectfully submitted,

Klifton L. Kime

Klifton L. Kime, Registration No. 42,733
Attorney for Appellant
USPTO Customer No. 66083

DORSEY & WHITNEY LLP
Republic Plaza Building, Suite 4700
370 Seventeenth Street
Denver, Colorado 80202-5647
Phone: (303) 629-3400
Fax: (303) 629-3450

IX. APPENDIX OF CLAIMS

1. (Previously Presented) A method for averaging two pixel values, comprising:

decoding a single machine code instruction comprising an address for a first input register, an address for a second input register, an address for an output register, an op code indicating a function to perform, and a rounding factor;

loading a plurality of first operands from the first input register;

loading a plurality of second operands from the second input register;

producing an average, based on the op code, of one of the plurality of first operands and one of the plurality of second operands, wherein the rounding factor indicates which of a plurality of rounding algorithms to use in producing the average, the plurality of rounding algorithms comprising:

a first rounding algorithm able to produce a change in the average; and

a second rounding algorithm able to produce a change in the average; and

storing the average in the output register.

2. (Original) The method of claim 1, determining how many fields are in each of the first and second input registers.

3. (Original) The method of claim 1, wherein the producing the average comprises:

producing a first intermediate result by adding one of the plurality of first operands to one of the plurality of second operands; and

producing the average by shifting the first intermediate result to the right by one binary digit.

4. (Previously Presented) The method of claim 1, wherein the producing the average comprises:

producing a first intermediate result by adding one of the plurality of first operands, one of the plurality of second operands and the rounding factor; and

producing the average by shifting the first intermediate result to the right by one binary digit.

5. (Original) The method of claim 1, further comprising rounding the average before storing the average.

6. (Previously Presented) The method of claim 1, further comprising:

evaluating the rounding factor; and

adding a value to the average.

7. (Previously Presented) The method of claim 6, wherein the value is one of zero and one.

8. (Previously Presented) A method for averaging two pixel values, comprising:

decoding a single machine code instruction indicating an address for a first input register, an address for a second input register, an address for an output register, an op code indicating a function to perform, and a rounding factor;

loading a first operand from an A1 field of the first input register;

loading a second operand from a B1 field of the second input register;

producing an average, based on the op code, of the first operand and the second operand, wherein the rounding factor indicates which of a plurality of rounding algorithms to use in producing the average, the plurality of rounding algorithms comprising:

a first rounding algorithm able to produce a change in the average; and

a second rounding algorithm able to produce a change in the average; and

storing the average in a C1 filed of the output register.

9. (Original) The method of claim 8, wherein the instruction is one of a plurality of instructions in a long instruction word.

10. (Original) The method of claim 8, determining how many fields are in each of the first and second input registers.

11. (Original) The method of claim8, wherein the producing an average comprises:

producing a first intermediate result by adding one of the plurality of first operands to one of the plurality of second operands; and

producing the average by shifting the first intermediate result to the right by one binary digit.

12. (Previously Presented) The method of claim 8, wherein the producing an average comprises:

producing a first intermediate result by adding one of the plurality of first operands, one of the plurality of second operands and the rounding factor; and

producing the average by shifting the first intermediate result to the right by one binary digit.

13. (Previously Presented) The method of claim 8, further comprising:

evaluating the rounding factor; and

adding a value to the average.

14. (Original) The method of claim 13, wherein the value is one of zero and one.

15. (Original) The method of claim 8, wherein the first input register comprises a plurality of fields.

16. (Original) The method of claim8, further comprising rounding the average before storing the average.

17. (Original) The method of claim 8, further comprising:

loading a third operand from an A2 field of the first input register;

loading a further operand from a B2 field of the second input register;

producing a second average of the third operand and the fourth operand; and

storing the second average in a C2 field of the output register.

18. (Previously Presented) A pixel averaging apparatus, comprising:

a first input register comprising a plurality of first fields;

a second input register comprising a plurality of second fields;

a rounding factor indicated by a single machine code instruction;

a plurality of average modules respectively coupled to the first and second fields, the modules configured to perform an averaging function indicated by an op code in the single machine code instruction; and

an output register comprising a plurality of third fields, wherein:

the third fields are respectively coupled to the plurality of average modules, and

the rounding factor affects how the plurality of average modules round results to produce an average,

wherein:

the single machine code instruction comprises an address for the first input register, an address for the second input register, an address for the output register, the op code, and the rounding factor; and

the rounding factor indicates which of a plurality of rounding algorithms to use in producing the average, the plurality of rounding algorithms comprising:

a first rounding algorithm able to produce a change in the average; and

a second rounding algorithm able to produce a change in the average.

19. (Original) The pixel averaging apparatus of claim 18, wherein the average module comprises:

a plurality of adders respectively coupled to the first and second fields; and

a plurality of shifters respectively coupled to the plurality of adders.

20. (Previously Present) The pixel averaging apparatus of claim 18, wherein the rounding factor causes at least one of rounding-up or rounding-down by the plurality of average modules.

21. (Previously Presented) The pixel averaging apparatus of claim 18, wherein the rounding factor is added to the first and second fields in the average module.

X. EVIDENCE APPENDIX

None.

XI. RELATED PROCEEDINGS APPENDIX

None.